## REMARKS

By the above amendment, Applicants have: 1) amended claims 1, 18, 21, 26, 31, 34, 51, 54, 59, and 64; 2) added no new claims; and 3) canceled no claims. As such, claims 1 – 69 are now pending. Support for the amendment is found in the specification, the drawings, and in the claims as originally filed. Applicants submit that the amendment does not add new matter. Applicants respectfully request reconsideration of the present application and consideration of the following remarks and the claims.

### Specification

***"The title of the invention is not descriptive. A new title is required that is clearly indicative of the invention to which the claims are directed."***

Applicants have amended the title according to the Examiner's suggestion as shown in page 2 of this amendment.

### Claim Objections

***"Claims listed below are objected to because of the following informalities: ..."***

Applicants wish to thank the Examiner for pointing out various informalities in claims 18 and 51, claims 21, 31, 54, and 64, and claims 26 and 59. Accordingly, proper corrections have been made to these claims.

## Claim Rejections - 35 U.S.C. § 102

***"Claims 1-2, 4-7, 9-20, 23-30, 32-35, 37-40, 42-53, 56-63, 65-66, and 68 are rejected under 35 U.S.C. 102(e) as being anticipated by Sazegari, U.S. Patent 6,446,198."***

The Office Action rejected claims 1-2, 4-7, 9-20, 23-30, 32-35, 37-40, 42-53, 56-63, 65-66, and 68 over Sazegari. Independent claims 1 and 34 have been amended. Applicants respectfully submit that these claims and all of their dependent claims in light of this amendment are patentable.

The amended claim 1 now recites:

1. (Currently Amended) A method for execution by a microprocessor in response to receiving a single instruction, the method comprising:

receiving a string of bits;

generating a plurality of indices using a plurality of segments of bits in the string of bits;

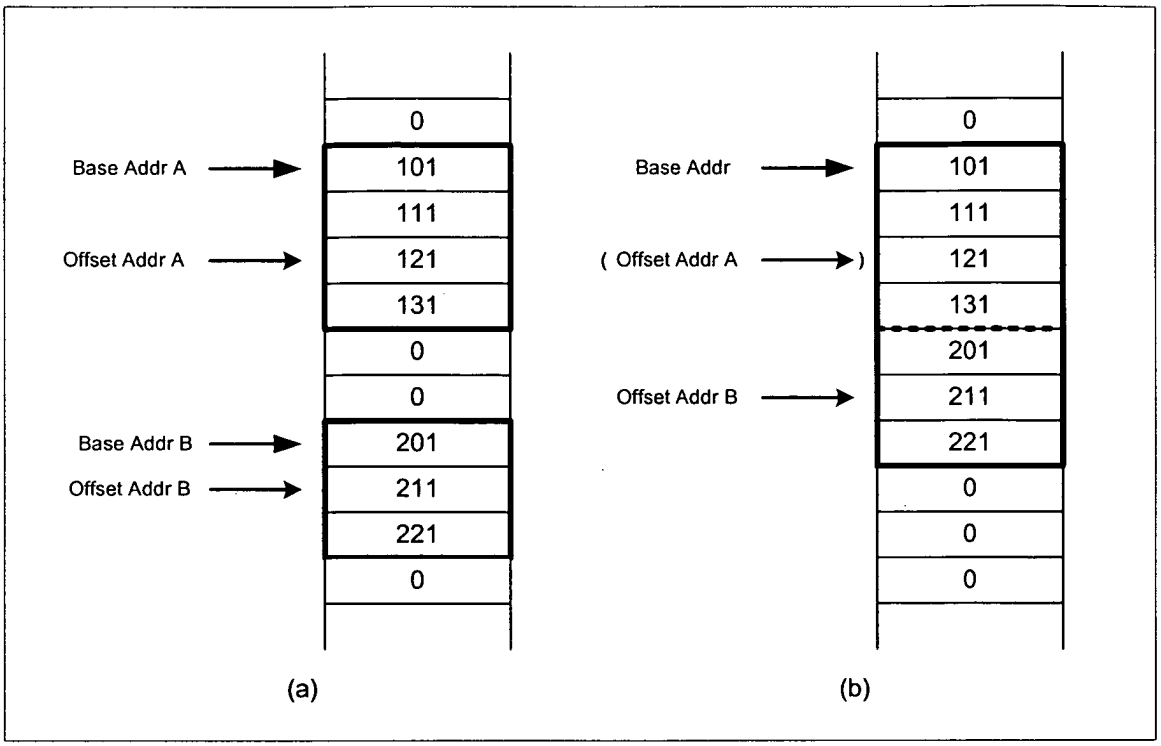
looking up simultaneously a plurality of entries from a plurality of look-up tables using the plurality of indices, wherein each of said plurality of look-up tables is separate and distinct from others of said plurality of look-up tables; and

combining the plurality of entries into a first result;

wherein the above operations are performed in response to the microprocessor receiving the single instruction.

The Office Action asserted that Sazegari showed each and every aspect of the present invention. In particular, it stated that Sazegari showed the limitation "plurality of look-up tables" because *"... the table of Fig. 4 is divided into two sub-tables (data1 and data2). ... a table is logically divided into a number of smaller tables (i.e., multiple logical tables exist)."*. Applicants respectfully disagree. The Office Action further stated that a table is defined to be *"an orderly arrangement of data, especially one in which the data are arranged in columns and rows in an essentially rectangular form"* citing Dictionary.com, and it concluded that the table/memory of Fig. 4 *"may be viewed as one table of 32 entries, 2 tables of 16 entries each, 4 tables of 8 entries each, and so on, until you have 32 tables of 1 entry each."* Applicants respectfully disagree. Words tend to have different, sometimes subtly different, meanings depending on the context in which they are used, and the commonly used English word "table" is no exception. Although the word "table" is used with more than one meanings in Sazegari, a person of ordinary skill in the art would have no problem recognizing different meanings depending on the contexts, and he would readily appreciate that the table of Fig. 4 is a single table[1] whether it is *logically* divided into multiple smaller tables[2] or sub-tables[2] or not. The word "table" in the first context (indicated by superscript 1) denotes a separate entity in the programming context, for example, implemented by a specific data structure such as an array or a hash table or any appropriate data types. On the other hand, "table" used informally in the second context (indicated by superscript 2) should be interpreted as a part of a table[1] (e.g. a rectangular region), which is not an independent entity and which has a meaning only in the context of its "parent" table[1].

For example, the following schematic drawing illustrates (a) two separate tables[1] and (b) one single table[1], which is logically divided into two sub-tables[2]. Suppose that these tables are implemented as array data structures in the C programming language, in which arrays are guaranteed to occupy contiguous blocks in a memory space.




Note that the tables in (a) and (b) contain the same elements. In C, an array is defined by its base address, and an offset is used to point to a specific element in the array. In figure (a), there are shown two base addresses since there are two tables[1], whereas in figure (b), there is only one base address. It should be noted that the pointer "Base Addr B" in (a) can be anywhere in a valid memory space and it is not in any way constrained to be located near "Base Addr A" since these two arrays are completely separate and independent entities. In (b), however, this *logical* division (indicated by a broken horizontal line between elements "131" and "201") does not require a separate additional pointer, and these two sub-tables[2] cannot be separated. (Note that, for example, in order to access the element, "211", the offset value 1 is used in (a), whereas the offset value is 5 for the element "211" in (b).)

In the example shown in Fig. 4 of Sazegari, the two data blocks, "data1" and "data2", of a table[1] should occupy contiguous area in the memory since these data are accessed by a single register as illustrated in the figure. People of ordinary skill in the art will appreciate, therefore, that this *logical* division shown in Fig. 4 of Sazegari does not divide the table[1] into two separate and independent entities. They will also recognize that this "logical division of a

table[1]" was used as a conceptual tool for the purposes of explaining various algorithms described in the reference.

Even though Applicants used an array in C in the above example to demonstrate the difference between tables[1] and tables[2], this argument is not limited to any particular data structures or any particular programming languages, or to any software or hardware platforms. The argument given here is for illustrative purposes only, and it should not be construed to limit any teachings of the present invention.

Regarding other teachings of Sazegari, Applicants respectfully submit that this point is indeed moot since the operations taught in Sazegari are done with a plurality of instructions, in particular, using multiple Permute operations and at least one Select operation.

In summary, Sazegari uses a plurality of instructions and uses only one table. Applicants therefore respectively submit that the amended claim 1 and its dependent claims 2-33 and 67-69 are patentable, at least for the above reasons, over the prior art references.

With regards to claim 34, the same argument as in claim 1 applies, and Applicants respectfully submit that the claim 34 and its dependent claims 35-66 are patentable as amended, and they request reconsideration of these claims.

## Claim Rejections - 35 U.S.C. § 103

***"Claims 3, 8, 21-22, 31, 36, 41, 54-55, 64, and 69 are rejected under 35 U.S.C. 103(a) as being unpatentable over Sazegari, as applied above."***

Regarding claims 3, 8, 21-22, 31, 36, 41, 54-55, 64, and 69, Applicants respectfully submit that these claims are patentable over Sazegari, at least for the same reasons given with respect to amended independent claims 1 and 34.

***"Claim 67 is rejected under 35 U.S.C. 103(a) as being unpatentable over Sazegari, as applied above, in view of Shams, U.S. Patent 5,526,501."***

Regarding claim 67, Applicants respectfully submit that this claim is patentable over Sazegari and Shams, at least for the same reasons given with respect to amended independent claim 1.

## CONCLUSION

For all the above reasons, Applicants submit that the specification and claims are now in proper form, and that the claims all define patentably over the prior art. Therefore they submit that all rejections have been overcome and that all pending claims are in condition for allowance, which action they respectfully solicit. If a telephone conference would facilitate the prosecution of this application, the Examiner is invited to contact Jimmi Yoon at (408) 720-8300, extension 305.

Authorization is hereby given to charge our Deposit Account No. 02-2666 for any charges that may be due or credit any overages. If an extension is required, Applicants hereby request such extension.

Respectfully Submitted,
BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP

Date: January 30th, 2006

H Yoon

Hyoungsoo "Jimmi" Yoon
Reg. No.: 57,637

12400 Wilshire Boulevard
Seventh Floor
Los Angeles, CA 90025-1030
(408) 720-8300